AMENDMENTS TO THE SPECIFICATION

Please replace the paragraph at page 4, lines 12-24, with the following amended paragraph:

--The present invention may advantageously provide detailed reports. Each time a build is scanned, the system may provide a report that lists differences tracked between a previously scanned software build. The report may indicate any new, changed, or deleted element or result. Additionally, the system may provide a report presenting information on how many classes were tested, including the number of [[block]] blocks within a class that were executed and the number of arcs followed in the execution path. This report may be used to drive future testing work and target areas of the code for additional testing. Furthermore, the present invention also provides code generation from intermediate code to specified targets to aid in reproducing and fixing bugs. This may also be useful--

Please replace the paragraph at page 14, lines 1-7, with the following amended paragraph:

--touch screen panel can be physically coupled to a housing in which the computing device 110 is incorporated, such as in a tablet-type personal computer. In addition, computers such as the computing device 110 may also include other peripheral output devices such as speakers [[195]] 194 and printer [[196]] 195, which may be connected through an output peripheral interface [[194]] 193 or the like.--

Please replace the paragraph at page 21, lines 1-10, with the following amended paragraph:

--A validator 414 may have children for storing information about a factory 418, a matrix generator 420, and a comparator 422. A validator may attach a validation pattern to a specific type in order to validate the type. A validator may contain a factory that you use to create types that you may pass as parameters. It may also contain a matrix generator to provide different combinations of parameter values for input as test vectors. A validator may also contain a comparator that may be used to verify that the actual output is the expected output.--

Please replace the paragraph at page 21, lines 21-24, with the following amended paragraph:

--FIG. 5 presents a flowchart generally representing example steps undertaken by the system for processing a software build. First, the system may [[checks]] check at step 504 if there is a configuration file containing additional--

Please replace the paragraph at page 26, lines 4-12, with the following amended paragraph:

--The system may then check at step 714 if that was the last type in the assembly. If it was not the last type, then the system may get the next type at step 704 and processing may continue. If it was the last type in the assembly, then the system may check whether this was the last assembly in the build at step 716. If it was not the last [[assemble]] assembly, the system may get the next assembly at step 702 and processing may continue until the last assembly may be processed. Otherwise, the system is finished processing.--

Please replace the paragraph at page 32, lines 10-16, with the following amended paragraph:

--If the type is not found in the old dictionary at step 1314, then the system may record the difference at step 1318. After recording any difference, the system may [[checks]] check if this is the last type in the new dictionary at step 1324. If this is the last type, then the system may check if this is the last assembly at step 1326. Otherwise, the system may select the next type at step 1308 and processing may continue.--

Please replace the paragraph at page 36, lines 18-24, with the following amended paragraph:

--FIG. 17 presents an illustration generally representing an example graphical user interface for reporting the status of a software build after processing by the system. The graphical user interface [[702]] 1702 may include a dictionary display area 1704, a query builder, 1706, an area for displaying attributes 1708, an area for displaying new types 1710 and an area for displaying bugs 1712.--